IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application of: Daniel Plastina et al. Art Unit 2452
Serial No. 10/623,235
Filed July 18, 2003
Confirmation No. 2732
For: METADATA RETRIEVAL PROTOCOLS AND NAMESPACE IDENTIFIERS
Examiner Tauqir Hussain

April 27, 2009

APPEAL BRIEF

Robert M. Bain, Reg. No. 36,736
SENNIGER POWERS LLP
100 North Broadway, 17th Floor
St. Louis, Missouri 63102
(314) 231-5400

TABLE OF CONTENTS

TABLE OF AUTHORITIES

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application of: Daniel Plastina et al. Art Unit 2452
Serial No. 10/623,235
Filed July 18, 2003
Confirmation No. 2732
For: METADATA RETRIEVAL PROTOCOLS AND NAMESPACE IDENTIFIERS
Examiner Tauqir Hussain

April 27, 2009

APPEAL BRIEF

This is an appeal from the final rejection of the claims of the above-referenced application made in the Final Office Action dated June 27, 2008. A Notice of Appeal was filed on November 25, 2008.

The appeal brief fee in the amount of $540.00 is submitted herewith.

I. REAL PARTY IN INTEREST

The real party in interest in connection with the present appeal is Microsoft Corporation of One Microsoft Way, Redmond, Washington, 98052, a corporation of the state of Washington, owner of 100 percent interest in the pending application.

II. RELATED APPEALS AND INTERFERENCES

Appellants are unaware of any pending appeals, interferences, or judicial proceedings that may be related to, directly affect or be directly affected by, or have a bearing on, the Board's decision in the pending appeal.

III. STATUS OF CLAIMS

Claims 1, 2, 5-7, 9-20, 22, 29, 30, 33-38, 40-43, 45-47, 50, 51, 53, 54, 64, 65, and 67, as set forth in the Claims Appendix, are currently pending in the application for consideration. Claims 3, 4, 8, 21, 23-28, 31, 32, 39, 44, 48, 49, 52, 55-63, 66, and 68-72 have been canceled.

Claims 1, 2, 5-7, 9-20, 22, 29, 30, 33-38, 40-43, 45-47, 50, 51, 53, 54, 64, 65, and 67 stand rejected. The rejection of each of these claims is being appealed.

IV. STATUS OF AMENDMENTS

Appellants filed Amendment D on September 29, 2008 subsequent to the final rejection made in the Final Office Action dated June 27, 2008. In Amendment D, Appellants amended claim 64 to comply with the requirement of form expressly set forth in the June 27, 2008 Final Office Action. The Advisory Action dated October 17, 2008 does not indicate whether the amendment to claim 64 will or will not be entered, but the Examiner indicated "OK TO ENTER: /TH/" on a copy of the first page of Amendment D. Appellants request entry of Amendment D for purposes of this appeal.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The following summary correlates claim elements to embodiments described in the application specification, but does not in any manner limit claim interpretation. Rather, the following summary is provided only to facilitate the Board's understanding of the subject matter of this appeal.

Aspects of the present application relate to obtaining metadata for media content. An exemplary consumer electronic device or media player 112 includes or has access to one or more computer-readable media having computer-executable components for obtaining metadata for a media content file. The media content file is stored on a computer-readable medium 110. In one embodiment, the metadata is available from a metadata provider 111 via a data communication network 113. See Application at paragraph [0044]; FIG. 1.

Independent claims 1, 29, 37, 43, 47, 51, and 64 are involved in the appeal.

Independent claim 1 is directed to a method for obtaining metadata associated with a media content file stored on a computer storage medium 110. For example, at step 202 of FIG. 2, a populated request data structure (MDQ) 602 requests metadata for the media content file from a metadata provider 111. As shown in FIG. 6, the request data structure 602 includes a request type identifier 604 defining a type for the computer-readable medium 110, a request identifier 606, and a plurality of metadata elements 608 stored with the media content file. See Application at paragraph [0060]; FIG. 6. As explained in the present application, a client computing device (e.g., media player 112) transmits the data structure 602 to the metadata provider 111 to request metadata for media content. See Application at paragraph [0060]. In one embodiment, the media player 112 generates a data request, as shown at step 304 of FIG. 3, for creating an HTTP GET at 312. See Application at paragraph [0055]; FIG. 3. In response to receiving the populated request data structure 602, the metadata provider 111 searches for the requested metadata in a database based on the received plurality of metadata elements 608 to identify relevant metadata from the search results. See Application at paragraphs [0047]-[0057]; FIGS. 2 and 6.

The method of claim 1 further includes receiving a return data structure, such as MDR 702 from the metadata provider 111, as shown at step 204 of FIG. 2. The return data structure 702 includes a return type identifier 704 defining the type for the computer-readable medium

110, the request identifier 606, identified relevant metadata 706 corresponding to the requested metadata, and a delay time interval (i.e., back off interval) 708. See Application at paragraph [0071]; FIG. 7. As explained in the application, the metadata provider 111 identifies the metadata relevant to or otherwise associated with the media content file as defined in the MDQ query (i.e., request data structure 602) and sends the identified metadata to the requesting client as the return metadata 706 defined in the MDR (i.e., return data structure 702) response. See Application at paragraphs [0047]-[0057]; FIGS. 2 and 7. As an example, metadata provider 111 generates an SQLXML query using the request identifier 606 from the HTTP GET, as shown at step 314 of FIG. 3, to query a database to obtain metadata associated with the identifier for delivery to the media player 112. See Application at paragraph [0056]; FIG. 3.

In addition, claim 1 recites postponing additional requests for metadata until after the delay time interval has elapsed. As described above, an exemplary return data structure 702 also includes delay time interval 708 to instruct the client to postpone additional requests for metadata until after the delay time interval 708 has elapsed for server load balancing reasons. See Application at paragraphs [0049], [0071].

Referring further to FIG. 2, at step 206, aspects of the application associate the return metadata 706 with namespace identifiers, including at least one of WMContentID, WMCollectionID, and WMCollectionGroupID (e.g., a box set identifier). The namespace identifiers and associated metadata are stored with the media content file at 208. Alternatively or in addition, the return metadata 706 and/or namespace identifiers are stored in a cache. Further, the client 112 may request additional metadata from the metadata provider 111 using a portion of the return metadata 706. Further aspects classify the media content based on the return metadata

706. For example, values are assigned to the identifiers WMPrimaryClassID, and WMSecondaryClassID. See Application at paragraph [0051].

In accordance with another aspect of the application, independent claim 29 recites one or more computer-readable media having computer-executable components for obtaining metadata for a media content file storing media content. In one embodiment, the components include a query component 114 for populating a request data structure (MDQ) 602, which includes a request type identifier 604 defining a type for the computer storage medium 110, a request identifier 606, and a plurality of metadata elements 608 stored with the media content file. See Application at paragraph [0060]; FIG. 6. The query component 114 requests metadata for the media content file from a metadata provider 111 via the populated request data structure 602. And, in response to receiving the populated request data structure 602, the metadata provider 111 searches for the requested metadata in a database based on the received plurality of metadata elements 608 to identify relevant metadata from the search. See Application at paragraph [0045]; FIG. 1.

In one embodiment, the components of claim 29 also include an interface component 116 for receiving a return data structure (MDR) 702 from the metadata provider 111 in response to the request sent by the query component 114. The return data structure 702 stores a delay time interval 708, a return type identifier 704 defining the type for the computer storage medium 110, the request identifier 606, and identified relevant metadata 706 corresponding to the requested metadata. See Application at paragraph [0071]; FIG. 7. In turn, the query component 114 postpones additional requests for metadata from the metadata provider 111 until after the delay time interval 708 has elapsed. See Application at paragraphs [0045], [0049]; FIG. 1.

Independent claim 37 relates to a media player 112 comprising computer-executable instructions for obtaining metadata for a media content file. These instructions, generally illustrated in FIG. 2 in one embodiment, include populating a request data structure (MDQ) 602, which includes a request type identifier 604 defining a type for the computer storage medium 110, a request identifier 606, and a plurality of metadata elements 608 stored with the media content file. See Application at paragraph [0060]; FIG. 6. The instructions further include, as shown at step 202 of FIG. 2, requesting metadata for the media content file from a metadata provider 111 via the populated request data structure 602. In response to receiving the populated request data structure 602, the metadata provider 111 searches for the requested metadata in a database based on the received plurality of metadata elements 608 to identify relevant metadata 706 from the search results, and correlates relevant metadata 706 from the search results to compute an accuracy rating based on the received plurality of metadata elements 608. See Application at paragraphs [0047]-[0057], [0063]; FIGS. 2 and 6.

The computer-executable instructions of claim 37 also include receiving a return data structure, such as MDR 702, including the accuracy rating, from the metadata provider 111, as shown at step 204 of FIG. 2. The return data structure 702 stores a delay time interval 708, a return type identifier 704 defining the type for the computer storage medium 110, the request identifier 606, and the identified relevant metadata 706 corresponding to the requested metadata. See Application at paragraph [0071]; FIG. 7. In addition, the instructions of claim 37 include postponing additional requests to the metadata provider 111 for metadata until after the delay time interval 708 has elapsed. See Application at paragraphs [0049], [0071].

Independent claim 43 recites a data structure (MDQ) 602 embodying further aspects. A first computing device (e.g., media player 112) transmits the data structure 602, which represents a request for metadata, to a second computing device (e.g., metadata provider 111). The data structure 602 includes a request type identifier 604 defining a type for a destination computer storage medium 110 storing the media content, a request identifier 606, and one or more metadata elements 608 stored with the media content. See Application at paragraph [0060]; FIG. 6. In this instance, the media content is one song from a plurality of songs associated with an album. See Application at paragraph [0050]. In response to the receipt of the data structure 602, the second computing device 111 returns metadata 706 for each of the plurality of songs associated with the album. See Application at paragraph [0074].

Independent claim 47 recites a data structure (MDR) 702 embodying further aspects. A first computing device (e.g. metadata provider 111) transmits the data structure 702 in response to a request for metadata from a second computing device (e.g., media player 112). The data structure 702 includes a return type identifier 704 defining a type for the destination computer storage medium 110 storing the media content, a request identifier 606, return metadata 706, and a delay time interval 708. See Application at paragraph [0071]; FIG. 7. In this instance, the media content is one song from a plurality of songs associated with an album. See Application at paragraph [0050]. In response to the metadata request, the first computing device 111 returns metadata 706 for each of the plurality of songs associated with the album. See Application at paragraph [0074]. Moreover, the second computing device (e.g., media player 112) postpones sending additional requests until after the delay time interval 708 has elapsed.

Referring now to one embodiment of independent claim 51, a computer storage medium has stored thereon a data structure 802 as shown in FIG. 8. The data structure 802, which represents a namespace for identifying media content, has a first field storing a content identifier value (WMContentID) 804 and a second field storing a collection identifier value

(WMCollectionID) 806. The content identifier value 804 is a GUID value representing a performance of a particular work embodied in the media content as it relates to a collection and the collection identifier value 806 is a GUID value representing a single physical medium of the collection. As set forth in the claim, the physical medium represented by the WMCollectionID 806 includes the performance represented by the WMContentID 804. The data structure 802 also includes a third field storing a group identifier value (WMCollectionGroupID) 808, which is a GUID value representing a plurality of physical mediums of the collection. The single physical medium represented by the WMCollectionID 806 is one of the plurality of physical mediums of the collection associated with the WMCollectionGroupID 808 and the first, second, and third fields represent increasing levels of granularity for characterizing the media content. See Application at paragraphs [0080]-[0084]; FIG. 8.

As described in paragraph [0085], WMCollectionGroupID 808 enables the media player 112 to display an accurate hierarchy in the media player 112 in instances where a specific album belongs to a multi-album set. This identifier represents various collections including, but not limited to, multiple-CD collections considered to be a single album, multiple-album collections (e.g., “box-sets”), which may include a multiple-CD collection, multiple-disc DVD collections (e.g., 2-disc movie releases), and multiple DVDs sold together as a single collection where each disc may include multiple discs. And, upon receipt of metadata 706 for media content, the media player 112 assigns three values (e.g., GUIDs) to the three identifiers WMContentID 804 (e.g., per track), WMCollectionID 806 (e.g., per album), and WMCollectionGroupID 808 (e.g., spans CDs). See Application at paragraph [0085].

Independent claim 64 recites a method for obtaining metadata for media content. For example, as shown generally at step 310 of FIG. 3, the method of claim 64 includes formulating a network address (e.g. URL) with a query string parameter. The query string parameter includes an identifier and a value associated therewith. The identifier, or a portion thereof, includes the text string WMID. For example, the WMID is the WMCollectionID 806 for the media content and the associated value corresponds to the media content. An exemplary URL http://windowsmedia.com/redir/GetMDRCD.asp?wmid=8C0E118D-36E7-43A0-8732-24FA851F8A80&version=9.0.0.0000&locale=409 &requestid=B1D119EA-4FC8-409F-BF05-D52E0FED2FDB illustrating the identifiers WMID, VERSION, LOCALE, and REQUESTID and their associated values. In claim 64, the media content file is one of a plurality of songs in an album. The method of claim 64 also includes requesting metadata for the media content file from a metadata provider 111 via the formulated network address and receiving a return data structure 702 from the metadata provider 111. The return data structure stores a return type identifier 704 defining a type for the computer storage medium 110, a request identifier 606, and return metadata 706 corresponding to the metadata for each of the plurality of songs in the album. See Application at paragraphs [0067]-[0070].

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

A. Appellants appeal the rejections of claims 1, 2, 5-7, 9-20, 22, 29, 30, 33-38, 40-43, 45-47, 50, 51, 53, 54, 64, 65, and 67 under 35 U.S.C. §103(a) as being unpatentable over Meyer et al., U.S. Published Application No. 2001/0031066, in view of Srivastava et al., U.S. Patent No. 6,549,922, and further in view of Berkun et al., U.S. Published Application No. 2002/0103920.

B. Appellants appeal the rejections of claims 53 and 54 under 35 U.S.C. §103(a) as being unpatentable over Meyer et al., U.S. Published Application No. 2001/0031066, in view of

Srivastava et al., U.S. Patent No. 6,549,922, further in view of Berkun et al., U.S. Published Application No. 2002/0103920, and further in view of Ramey, U.S. Published Application No. 2004/0059795.

C. Appellants appeal the objection to the specification under 37 CFR 1.75(d)(1) and MPEP 608.01(o) for failing to provide proper antecedent basis for the claimed subject matter.

D. Appellants appeal the objection to the drawings under 37 CFR 1.83(a) for failing to show every feature of the invention specified in the claims.

VII. ARGUMENT

A. Claims 1, 2, 5-7, 9-20, 22, 29, 30, 33-38, 40-43, 45-47, 50, 51, 53, 54, 64, 65, and 67 are patentable under 35 U.S.C. §103(a) as being nonobvious over the Meyer, Srivastava, and Berkun references.

B. Claims 53 and 54 are patentable under 35 U.S.C. §103(a) as being nonobvious over the Meyer, Srivastava, Berkun, and Ramey references.

C. The specification provides proper antecedent basis for the claimed subject matter in compliance with 37 CFR 1.75(d)(1) and MPEP 608.01(o).

D. The drawings show every feature of the invention specified in the claims in compliance with 37 CFR 1.83(a).

A. CLAIMS 1, 2, 5-7, 9-20, 22, 29, 30, 33-38, 40-43, 45-47, 50, 51, 53, 54, 64, 65, and 67 ARE NONOBVIOUS UNDER 35 U.S.C. §103(a) AND PATENTABLE OVER MEYER ET AL., SRIVASTAVA ET AL., AND BERKUN ET AL.

The independent claims, as amended, are allowable because the cited art does not make obvious at least (1) postponing additional requests for metadata from a metadata provider until after a delay time interval has elapsed (as recited by independent claims 1, 29, 37 and 47);

(2) submitting a request for metadata for a song associated with an album and receiving metadata for each song associated with the album (as recited by independent claims 43 and 64); or

(3) representing increasing levels of granularity for characterizing the media content using three fields of a data structure (as recited by independent claim 51).

Meyer et al. disclose transforming media objects, namely, media signals in electronic form, into active, connected objects via identifiers embedded therein. See Meyer et al., Abstract. As taught by the Meyer reference, an identifier associated with each media object (e.g., each audio file) is extracted and sent to a server that maps the identifier to an action (e.g., returning metadata). Specifically, Meyer et al. disclose a number of ways to associate an identifier with an audio object and describes means for encoding and decoding the identifier. See Meyer et al., paragraphs [0013]-[0014].

The Srivastava patent teaches the automatic extraction and transformation of metadata into logical annotations. Srivastava et al., Abstract. Srivastava et al. disclose storing both the media and its associated XML document containing the annotations in a database. Srivastava et al., column7, lines 63-67; column 8, lines 27-36.

Berkun et al. teach a method for calculating a relevancy score by a full-text relevancy ranker. See Berkun et al., FIG. 11. The relevancy score is based on categorized metadata and used by the relevancy ranker to rank documents returned in search results. See Berkun et al., paragraphs [0075]-[0076].

As explained in greater detail below, these references, whether considered separately or together, fail to teach or suggest each and every element of Appellants' claims.

Postponing Additional Requests For Metadata From A Metadata Provider Until After A Delay Time Interval Has Elapsed - *Claims 1, 29, 37, and 47*

Claim 1 recites a method for obtaining metadata associated with a media content file. In the claimed method, a populated request data structure requests metadata for the media content file from a metadata provider. The request data structure includes a request type identifier defining a type for the computer-readable medium on which the media content file is stored, a request identifier, and a plurality of metadata elements stored with the media content file. In response to receiving the populated request data structure, the metadata provider searches for the requested metadata in a database based on the received plurality of metadata elements to identify relevant metadata from the search results. The method of claim 1 further includes receiving a return data structure from the metadata provider. The return data structure includes a return type identifier defining the type for the computer-readable medium, the request identifier, identified relevant metadata corresponding to the requested metadata, and **a delay time interval** (i.e., back off interval). With respect to the delay time interval, claim 1 recites **postponing additional requests for metadata until after the delay time interval has elapsed**.

On page 7 of the June 27, 2008 Final Office Action, the Examiner admits that the Meyer and Srivastava references are silent with respect to these aspects of the claims. The Examiner asserts, however, that paragraph [0038] of the Berkun publication teaches the delay time interval recited in the claims. But Berkun et al. merely teach a system where a spider is used to locate media URLs that are passed to an extraction queue. Each item in the queue is assigned a processing time and a priority. In the exemplary embodiment, **each queue entry is given a processing time of now and the same default priority**. Furthermore in paragraph [0041], the Berkun reference teaches that if a media file and metadata links are invalid or inaccessible, the same object is re-queued and assigned a later time and priority.

This is completely different than the claimed delay time interval and postponing additional metadata request until after the interval has elapsed. First, Berkun et al. teach **re-queuing** occurs when a media file and metadata links are invalid or inaccessible. In contrast, claim 1 recites "**receiving a return data structure from the metadata provider, said return data structure storing ... a delay time interval**". In other words, the Berkun reference does not disclose storing this delay time interval, or back off interval, in a return data structure.

Second, Berkun et al. teach the same object is re-queued. And, even if re-queuing requests for invalid or inaccessible media files and metadata links creates a delay, other requests queued with a processing time of "now" will continue to be processed. In contrast, claim 1 recites "**postponing additional requests for metadata until after the delay time interval has elapsed**."

Third, the delay time interval of claim 1 is not equivalent to the processor processing request one at a time as asserted by the Examiner on pages 2, 3, and 8 of the June 27, 2008 action. **The claim recites postponing additional requests for metadata until after the delay time interval has elapsed.** As shown in exemplary MDR-CD **return data structure** in paragraph [0072] of the present application and reproduced in-part below, the delay time interval (e.g., backoff) is specified by the return data structure itself.

```
<Backoff>
    <Time>5</Time>
 </Backoff>
```

And, as explained above, additional requests are **postponed until after the delay time interval has elapsed**. Advantageously, **the delay time interval** may be used to instruct the client to postpone **additional requests for metadata** until after the delay time interval has elapsed **for server load balancing reasons**.

The cited references, separately or in combination with the other cited references, do not teach or suggest **receiving a return data structure that stores a delay time interval** and **postponing additional requests for metadata until after the delay time interval has elapsed** as recited in claim 1. The Examiner admits that the Meyer and Srivastava references are silent with respect to these aspects of the claims and Appellants submit the Berkun reference cannot cure their deficiencies for the reasons set forth above. Thus, claim 1 is allowable and the rejection should be withdrawn. Independent claims 29, 37, and 47 similarly recite a delay time interval and postponing additional requests for metadata until after this interval has elapsed and are believed to be allowable for at least the same reasons as claim 1. Claims 2, 5-7, 9-20, and 22, claims 30 and 33-36, claims 38 and 40-42, and claim 50 depend from claims 1, 29, 37, and 47, respectively, and are allowable for at least the same reasons as the independent claim from which they depend.

Submitting A Request For Metadata For A Song - *Claims 43 and 64*

Independent claim 43 recites a data structure embodying further aspects of the invention. A first computing device (e.g., a media player) transmits the data structure to a second computing device (e.g., a metadata provider) to request metadata. As set forth in the claim, the data structure includes a request type identifier defining a type for a destination computer storage medium storing the media content, a request identifier, and one or more metadata elements stored with the media content. In this instance, **the media content is one song from** a plurality of songs associated with **an album**. In response to the receipt of the data structure, the second computing device **returns metadata for each of the plurality of songs associated with the album**.

On page 3 of the June 27, 2008 Final Office Action, the Examiner asserts paragraphs [0030]-[0032] of the Berkun publication teach returning metadata for each of the plurality of songs associated with an album when metadata for a single song from the album is requested. Appellants disagree. Berkun et al. teach in paragraph [0030] that **media files and related metadata are searched for and retrieved by reading, extracting, enhancing, and grouping metadata** describing the contents for files. Paragraph [0031] of this reference teaches that upon finding a media file, metadata associated with that file is extracted. And, in paragraph [0032], Berkun et al. teach the extracted metadata is enhanced. In particular, paragraph [0032] discloses that **if metadata associated with a song comprises the fields of Composer, Title, Musician, Album Name, and Music Genre, but is missing the date the song was copyrighted, the copyright date is added to the extracted metadata**.

Additionally, on page 11 of the June 27, 2008 action, the Examiner asserts the Srivastava reference teaches returning metadata for each of the plurality of songs associated with an album when metadata for a single song from the album is requested. But Srivastava et al. describe, at column 8, lines 37-52, mapping the metadata from XML documents into a corresponding schema used by a database for storing, indexing, searching, and managing media and its metadata. With respect to the table shown in column 8 of the Srivastava reference, which lists a possible predefined set of media notations (e.g. metadata), the cited art explains **not all media fields will provide values for every attribute in the predefined set**. See Srivastava, column 6, lines 22-26 and column 7, lines 27-30.

Appellants submit that the cited art discloses something entirely different than "a request type identifier defining a type for a destination computer storage medium storing the media content, said media content being **one song from a plurality of songs associated with an album** . . . wherein, **in response to the receipt of the data structure**, the second computing device returns metadata for **each of the plurality of songs associated with the album**" as recited in claim 43. In other words, when a client requests metadata for a track (e.g., song), the metadata provider returns metadata for a complete album. For example, even though the request pertained to metadata associated with a particular album track, the client device (e.g., a media player) may store the album information in a local cache. And, responsive to a subsequent request for metadata associated with another track of the album, the client device may retrieve the metadata from the local cache instead of the metadata provider. Advantageously, if CDs have an average of fifteen tracks, this data structure improves performance by greater than fifteen times for users who have full CDs. See Application, paragraph [0074].

Thus, the Berkun and Srivastava references, alone or in combination with the other cited references, do not teach or suggest a request including **metadata elements stored with the media content being <u>one</u> song from a plurality of songs associated with an album** and **returning metadata for the <u>each of the plurality</u> of songs associated with the album** as recited in claim 43. Claim 64 recites "receiving a return data structure from the metadata provider, said return data structure storing . . . return metadata corresponding to the metadata for **each of the plurality of songs in the album**" in response to a query for metadata associated with one of a plurality of songs in an album and, thus, is believed to be allowable for at least the same reasons as claim 43. Appellants submit claims 43 and 64 are allowable and the rejections should be withdrawn. Claims 45 and 46 and claims 65 and 67 depend from claims 43 and 64, respectively, and are allowable for at least the same reasons as the independent claims from which they depend.

Representing Increasing Levels Of Granularity For Characterizing The Media Content -- *Claim 51*

Referring now to independent claim 51, a data structure, which represents a namespace for identifying media content, has a first field storing a content identifier value (WMContentID) and a second field storing a collection identifier value (WMCollectionID). The content identifier value is a GUID value representing a performance of a particular work embodied in the media content as it relates to a collection and the collection identifier value is a GUID value representing a single physical medium of the collection. As set forth in the claim, the physical medium represented by the WMCollectionID includes the performance represented by the WMContentID. The data structure also includes a third field storing a group identifier value (WMCollectionGroupID), which is a GUID value representing a plurality of physical mediums of the collection. The single physical medium represented by the WMCollectionID is one of the plurality of physical mediums of the collection associated with the WMCollectionGroupID and the first, second, and third fields represent **increasing levels of granularity for characterizing the media content**.

On page 17 of the June 27, 2008 Final Office Action, the Examiner asserts the Berkun reference teaches WMContentID, WMCollectionID, and WMCollectionGroupID as recited in claim 51. Appellants disagree. Paragraphs [0004], [0043], and [0044] of the Berkun reference merely discuss streaming media concepts generally and the use of a spider for finding a media file containing a song and then extracting, parsing, and indexing metadata into several database fields for comparison to a known database. Berkun et al. teach the fields of referring URL, media URL, title, and performer.

The recited namespace identifiers of claim 51 are not merely a design choice. Instead, the namespace identifiers represent increasing levels of granularity for classifying the media content. In particular, WMContentID represents **a performance of a particular work as it relates to a collection**, WMCollectionID represents **a single physical medium of the collection wherein the physical medium represented by the WMCollectionID includes the performance represented by the WMContentID**, and WMCollectionGroupID represents **a plurality physical mediums of the collection** (e.g., box set).

As described in paragraph [0085] of the present application, WMCollectionGroupID enables the media player to display an accurate hierarchy in the media player where a specific album belongs to a multi-album set. This identifier represents various collections including, but not limited to, multiple-CD collections considered to be a single album, multiple-album collections (e.g., "box-sets"), which may include a multiple-CD collection, multiple-disc DVD collections (e.g., 2-disc movie releases), and multiple DVDs sold together as a single collection where each disc may include multiple discs. And, upon receipt of the return metadata for the media content, the media player assigns three values (e.g., GUIDs) to the three identifiers WMContentID (e.g., per track), WMCollectionID (e.g., per album), and WMCollectionGroupID (e.g., spans CDs).

In contrast, the cited art is silent with respect to this aspect of granularity. Nothing in the references teaches or suggests these three fields of increasing granularity and, especially, the third field in which WMCollectionGroupID represents **a plurality physical mediums of the collection**. Because the cited art fails to teach or suggest this aspect, Appellants submit claim 51 is allowable. Claims 53 and 54 depend from claim 51 and are allowable for at least the same reasons as claim 51.

B. CLAIMS 53 and 54 ARE NONOBVIOUS UNDER 35 U.S.C. §103(a) AND PATENTABLE OVER MEYER, SRIVASTAVA, BERKUN, AND RAMEY

Claims 53 and 54 depend from claim 51, and for at least the reasons above, Appellants respectfully submit that claims 53 and 54 are also patentable over the cited art. Hence, the rejection of claims 53 and 54 under 35 U.S.C. §103(a) should be withdrawn.

C. THE SPECIFICATION PROVIDES PROPER ANTECEDENT BASIS FOR THE CLAIMED SUBJECT MATTER

The specification stands objected to as failing to provide proper antecedent basis for the claimed subject matter. In particular, the Examiner asserts he could not find a support in the specification for the following subject matter of claim 37: "data structure storing a delay time interval." Appellants submit the subject matter of claim 37 is supported at least on page 11, paragraph [0049] of the application as originally filed, which recites "the **return data structure may also include a delay time interval** (e.g., a backoff interval) to instruct the client to postpone additional requests for metadata until after the delay time interval has elapsed for server load balancing reasons." Additionally, the exemplary MDR-CD **return data structure** described at pages 20-21, paragraph [0072] and illustrated in FIG. 7 of the present application includes the delay time interval (e.g., back off interval 708). Additional support can be found at, for example, pages 24-25, paragraphs [0077]-[0079]. According to MPEP 608.01(o), "an applicant is not limited to the nomenclature used in the application as filed [but] he or she should make appropriate amendment of the specification whenever this nomenclature is departed from by amendment of the claims so as to have clear support or antecedent basis in the specification for the new terms appearing in the claims." In light of the foregoing, the nomenclature of the claims does not depart from that of the specification and the meanings of the claims are readily ascertainable by one of ordinary skill in the art when read in light of the specification. Appellants submit the claims comply with MPEP 608.01(o) and 37 CFR 1.75(d)(1) and request the objection to the specification be withdrawn. In the event the Board believes that the claim language needs clarification, Appellants request that the Board remand the application to the Examiner to permit Appellants an opportunity to amend the claims to resolve any minor claim language issues.

D. THE DRAWINGS SHOW EVERY FEATURE OF THE INVENTION SPECIFIED IN THE CLAIMS

The drawings stand objected to under 37 CFR 1.83(a) for not showing every feature of the invention specified in the claims. Specifically, the Examiner asserts that the "additional requests for media until after the delay time interval has elapsed" feature in claims 1, "postponing additional requests for metadata from metadata provider until after the delay time interval has elapsed" in claim 29, "including a delay time interval, wherein the second computing device postpones sending additional requests until after the delay time interval has elapsed" in claim 47 and "data structure storing a delay time interval" or "postponing additional requests for metadata from metadata provider until after the delay time interval has elapsed" in claim 37 must be shown.

Appellants submit that FIG. 7, which illustrates an exemplary MDR data structure, includes reference character 708, "back off interval." According to 37 CFR 1.83(a), graphical drawing symbols or labeled representations may be used where detailed illustration is not essential for a proper understanding of the invention. The present application discloses on page 19, paragraph [0071] that the MDR data structure further includes **a back off interval 708** or other delay interval **specifying a time period for postponing additional requests for metadata by the second computing device**. And the application discloses at page 11 paragraph 49, for example, that "the **return data structure may also include a delay time interval** (e.g., a back off interval) to instruct the client to postpone additional requests for metadata until after the delay time interval has elapsed for server load balancing reasons." When considered by one of ordinary skill in the art, at least the data structure of FIG. 7 provides ample support in the drawings for the delay time interval claim elements. Thus, Appellants submit the drawings show every feature of the invention specified in the claims and request the objection to the drawings be withdrawn. In the event the Board believes that the drawings require clarification or correction, Appellants request that the Board remand the application to the Examiner to permit Appellants an opportunity to file amended drawings to resolve any minor drawing issues.

VIII. CONCLUSION

For the reasons stated above, Appellants respectfully request that the Office's rejections be reversed and that claims 1, 2, 5-7, 9-20, 22, 29, 30, 33-38, 40-43, 45-47, 50, 51, 53, 54, 64, 65, and 67 be allowed.

Respectfully submitted,

/Robert M. Bain/

Robert M. Bain, Reg. No. 36,736
SENNIGER POWERS LLP
100 North Broadway, 17th Floor
St. Louis, Missouri 63102
(314) 345-7000

RMB/lav

IX. CLAIMS APPENDIX

Claim 1 (previously presented): A method for obtaining metadata for a media content file storing media content, said media content file being stored on a computer storage medium, said method comprising:

populating a request data structure, said request data structure comprising a request type identifier defining a type for the computer storage medium, a request identifier, and a plurality of metadata elements stored with the media content file;

requesting metadata for the media content file from a metadata provider via the populated request data structure, wherein, in response to receiving the populated request data structure, the metadata provider searches for the requested metadata in a database based on the received plurality of metadata elements and identifies relevant metadata from the search results;

receiving a return data structure from the metadata provider, said return data structure storing a return type identifier defining the type for the computer storage medium, the request identifier, identified relevant metadata corresponding to the requested metadata, and a delay time interval; and

postponing additional requests for metadata until after the delay time interval has elapsed.

Claim 2 (original): The method of claim 1, wherein the return metadata comprises metadata determined by the metadata provider to be associated with the media content file.

Claims 3 and 4 (canceled).

Claim 5 (original): The method of claim 1, wherein the type relates to at least one of the following: a compact disc, a digital versatile disc, and flash memory.

Claim 6 (previously presented): The method of claim 1, wherein the computer storage medium comprises one or more of the following: a compact disc, a digital versatile disc, and flash memory.

Claim 7 (previously presented): The method of claim 1, wherein the metadata provider comprises a computer.

Claim 8 (canceled).

Claim 9 (original): The method of claim 1, further comprising:

associating the return metadata or a portion thereof with namespace identifiers including at least one of WMContentID, WMCollectionID, and WMCollectionGroupID; and

storing the namespace identifiers and associated metadata with the media content file.

Claim 10 (original): The method of claim 9, wherein the return metadata comprises a globally unique identifier.

Claim 11 (original): The method of claim 1, further comprising classifying the media content with namespace identifiers including at least one of WMPrimaryClassID and WMSecondaryClassID.

Claim 12 (original): The method of claim 1, further comprising associated the return metadata or a portion thereof with a namespace identifier representing a box set identifier.

Claim 13 (original): The method of claim 1, wherein the metadata elements in the request data structure comprise values associated with namespace identifiers including at least one of WMContentID, WMCollectionID, WMCollectionGroupID, WMPrimaryClassID, and WMSecondaryClassID, wherein the values and namespace identifiers are stored in the media content file.

Claim 14 (original): The method of claim 13, wherein requesting the metadata comprises requesting the metadata from at least one of the following: a local cache, a network server, and a client computer.

Claim 15 (previously presented): The method of claim 1, wherein the media content file comprises one of a plurality of songs in an album, wherein requesting the metadata comprises requesting metadata for the song included in the media content file, and wherein the return metadata comprises metadata for the plurality of songs in the album at least one of the plurality of songs not included in the media content file.

Claim 16 (original): The method of claim 1, further comprising storing the return metadata in a cache.

Claim 17 (original): The method of claim 1, further comprising storing the return metadata with the media content file.

Claim 18 (original): The method of claim 1, further comprising requesting additional metadata from the metadata provider using a portion of the return metadata.

Claim 19 (original): The method of claim 1, wherein requesting the metadata comprises formulating a network address with one or more query string parameters, said formulated network address representing the request data structure.

Claim 20 (original): The method of claim 1, wherein the network address comprises a uniform resource locator.

Claim 21 (canceled).

Claim 22 (previously presented): One or more computer storage media having computer-executable instructions for performing the method of claim 1.

Claims 23-28 (canceled).

Claim 29 (previously presented): One or more computer storage media having computer-executable components for obtaining metadata for a media content file storing media content, said media content file being stored on a computer storage medium, said components comprising:

a query component for populating a request data structure, said request data structure comprising a request type identifier defining a type for the computer storage medium, a request identifier, and plurality of metadata elements stored with the media content file, said query component further requesting metadata for the media content file from a metadata provider via the populated request data structure, wherein, in response to receiving the populated request data structure, the metadata provider searches for the requested metadata in a database based on the received plurality of metadata elements and identifies relevant metadata from the search; and

an interface component for receiving a return data structure from the metadata provider in response to the request sent by the query component, said return data structure storing a delay time interval, a return type identifier defining the type for the computer storage medium, the request identifier, and identified relevant metadata corresponding to the requested metadata, wherein the query component postpones additional requests for metadata from the metadata provider until after the delay time interval has elapsed.

Claim 30 (previously presented): The computer storage media of claim 29, wherein the return metadata comprises metadata determined by the metadata provider to be associated with the media content file.

Claims 31-32 (canceled).

Claim 33 (previously presented): The computer storage media of claim 29, further comprising an authoring component for:

associating the return metadata or a portion thereof with namespace identifiers including at least one of WMContentID, WMCollectionID, and WMCollectionGroupID; and

storing the namespace identifiers and associated metadata with the media content file.

Claim 34 (previously presented): The computer storage media of claim 33, wherein the authoring component further classifies the media content using other namespace identifiers including at least one of WMPrimaryClassID and WMSecondaryClassID.

Claim 35 (previously presented): The computer storage media of claim 33, wherein the authoring component further comprises:

determining an identifier value;

associating the determined identifier value with media content; and

assigning the determined identifier value to one or more of the following namespace identifiers: WMContentID, WMCollectionID, and WMCollectionGroupID; and

storing the identifier value and assigned namespace identifiers with the media content.

Claim 36 (previously presented): The computer storage media of claim 29, wherein the metadata elements in the request data structure comprise values associated with namespace identifiers including at least one of WMContentID, WMCollectionID, WMCollectionGroupID, WMPrimaryClassID, and WMSecondaryClassID, wherein the values and namespace identifiers are stored in the media content file.

Claim 37 (previously presented): A media player comprising computer-executable instructions for obtaining metadata for a media content file, said media content file being stored on a computer storage medium, said instructions comprising:

populating a request data structure, said request data structure comprising a request type identifier defining a type for the computer storage medium, a request identifier, and a plurality of metadata elements stored with the media content file;

requesting metadata for the media content file from a metadata provider via the populated request data structure, wherein, in response to receiving the populated request data structure, the metadata provider searches for the requested metadata in a database based on the received plurality of metadata elements, identifies relevant metadata from the search results, and correlates relevant metadata from the search results to compute an accuracy rating based on the received plurality of metadata elements;

receiving a return data structure including the accuracy rating from the metadata provider, said return data structure storing a delay time interval, a return type identifier defining the type for the computer storage medium, the request identifier, and the identified relevant metadata corresponding to the requested metadata; and

postponing additional requests for metadata from the metadata provider until after the delay time interval has elapsed.

Claim 38 (original): The media player of claim 37, wherein the instructions further comprise classifying the media content file based on the return metadata.

Claim 39 (canceled).

Claim 40 (original): The media player of claim 37, wherein the instructions further comprise: associating the return metadata or a portion thereof with namespace identifiers including at least one of WMContentID, WMCollectionID, WMCollectionGroupID; and

storing the namespace identifiers and associated metadata with the media content file.

Claim 41 (original): The media player of claim 37, wherein the instructions further comprise classifying the media content using other namespace identifiers including at least one of the following: WMPrimaryClassID and WMSecondaryClassID.

Claim 42 (original): The media player of claim 37, wherein the instructions further comprise:

determining an identifier value;

associating the determined identifier value with media content; and

assigning the determined identifier value to one or more of the following namespace identifiers: WMContentID, WMCollectionID, and WMCollectionGroupID; and

storing the identifier value and assigned fields with the media content.

Claim 43 (previously presented): A computer storage medium having stored thereon a data structure representing a request for metadata, said data structure for transmission by a first computing device to a second computing device to request metadata for media content, said data structure comprising:

a request type identifier defining a type for a destination computer storage medium storing the media content, said media content being one song from a plurality of songs associated with an album;

a request identifier; and

one or more metadata elements stored with the media content, wherein, in response to the receipt of the data structure, the second computing device returns metadata for each of the plurality of songs associated with the album .

Claim 44 (canceled).

Claim 45 (previously presented): The computer storage medium of claim 43, wherein the type relates to at least one of the following: a compact disc, a digital versatile disc, and flash memory.

Claim 46 (previously presented): The computer storage medium of claim 43, wherein the destination computer storage medium comprises one or more of the following: a compact disc, a digital versatile disc, and flash memory.

Claim 47 (previously presented): A computer storage medium having stored thereon a data structure sent from a first computing device to a second computing device in response to a request for metadata sent by the second computing device, said data structure comprising:

a return type identifier defining a type for a destination computer storage medium storing the media content, said media content being one song from a plurality of songs associated with an album, ;

a request identifier; and

return metadata for the plurality of songs associated with the album corresponding to the requested metadata including a delay time interval, wherein the second computing device postpones sending additional requests until after the delay time interval has elapsed.

Claims 48-49 (canceled).

Claim 50 (previously presented): The computer storage medium of claim 47, wherein the type relates to at least one of the following: a compact disc, a digital versatile disc, and flash memory.

Claim 51 (previously presented): A computer storage medium having stored thereon a data structure representing a namespace for identifying media content, said data structure comprising:

a first field storing a content identifier value, said first field having a label of WMContentID, said content identifier value being a GUID value representing a performance of a particular work as it relates to a collection, said performance being embodied in the media content;

a second field storing a collection identifier value, said second field having a label of WMCollectionID, said collection identifier value being a GUID value representing a single physical medium of the collection wherein the physical medium represented by the WMCollectionID includes the performance represented by the WMContentID; and

a third field storing a group identifier value, said third field having a label of WMCollectionGroupID, said group identifier value being a GUID value representing a plurality physical medium of the collection, wherein the single physical medium represented by the WMCollectionID is one of the plurality of physical medium of the collection associated with the WMCollectionGroupID and said first, second, and third fields represent increasing levels of granularity for characterizing the media content.

Claim 52 (canceled).

Claim 53 (previously presented): The computer storage medium of claim 51, wherein the content identifier value, the collection identifier value, and the group identifier value each comprise a globally unique identifier.

Claim 54 (previously presented): The computer storage medium of claim 51, wherein the third field represents a box set identifier.

Claims 55-63 (canceled).

Claim 64 (currently amended): A method for obtaining metadata for media content, said media content being stored on a computer storage medium, said method comprising:

formulating a network address with a query string parameter, said query string parameter comprising an identifier and a value associated therewith, said identifier or a portion thereof comprising the text string WMID, said associated value corresponding to the media content, wherein the media content file comprises one of a plurality of songs in an album;

requesting metadata for the media content file from a metadata provider via the formulated network address; and

receiving a return data structure from the metadata provider, said return data structure storing a return type identifier defining a type for the computer storage medium, a request identifier, and return metadata corresponding to the metadata for each of the plurality of songs in the album.

Claim 65 (original): The method of claim 64, wherein the formulated network address comprises a uniform resource locator.

Claim 66 (canceled).

Claim 67 (original): The method of claim 64, further comprising another query string parameter, said query string parameter comprising another identifier and another value associated therewith, said other identifier comprising one or more of the following: VERSION, LOCALE, and REQUESTID.

Claims 68-72 (canceled).

X. EVIDENCE APPENDIX

None.

XI. RELATED PROCEEDINGS APPENDIX

None.